

NOY - TP 1 : Ordonnancement et synchronisation

Par Léo Cosseron et Lendy Mulot

Où trouver le code

Notre code est disponible sur <https://github.com/ZeGmX/NOY-TP>. La version finale du TP est celle du commit `e17eceaa371ff727f89e98c89d4ddd324a96ee23`.

Les fichiers modifiés pour ce TP sont les suivants :

- `kernel/synch.cc`
- `kernel/exception.cc`
- `kernel/thread.cc`

Les fichiers ajoutés sont les suivants :

- `test/test_semaphore.c`
- `test/test_lock.c`
- `test/test_cond_var.c`

Avancement du TP

Toutes les parties obligatoires du TP ont été implémentées, et nos tests s'exécutent sans erreur. Aucun bonus n'a été implémenté pour le moment, nous reviendrons dessus une fois les trois TP obligatoires terminés.

Tests

Nous avons implémenté trois fichiers de test, présents dans le dossier `test/` :

- `test_semaphore.c` implémente une situation de producteur/consommateur, utilisant trois sémaphores et leurs primitives de synchronisation, pour deux threads.
- `test_lock.c` implémente une situation d'accès à un *buffer* en exclusion mutuelle.
- `test_cond_var.c` implémente un test des fonctions `CondBroadcast` et `CondWait` des variables de condition. Quatre *threads* sont en attente sur une condition, tandis qu'un cinquième attend que l'utilisateur entre un '1' pour réveiller les autres *threads*.

les trois fichiers permettent au passage de tester le bon fonctionnement des *threads*.